WHAT IS CLAIMED IS:

Gub A10

3

4

1. A method comprising:

identifying portions of a model as being either critical to a real-time execution of the model or non-critical to a real-time execution of the model; and

- generating code that is capable of real-time execution based on the critical portions of the model.
- 2. The method of claim 1 wherein non-critical portions are postprocessing units.
- 3. The method of claim 2 wherein post-processing units are logical units of the model that have no data outputs that feed non-post-processing sections of the model.
 - 4. The method of claim 1 wherein generating further comprises establishing an inter-process communication link between the code and the non-critical portions of the model.
- 5. The method of claim 4 further comprising receiving output from the code via the inter-process communications link.
- 1 6. The method of claim 5 further comprising executing the code on a target processor.
- 7. The method of claim 5 further comprising processing the output in the non-critical portions of the model.
- 8. A computer program product residing on a computer readable medium having instructions stored thereon which, when executed by
- a processor, cause the processor to:

2

3

5

6

1

3

5

4

5

6

8

identify portions of a model as being either critical to a real-time execution of the model or non-critical to a real-time execution of the model; and

generate code that is capable of real-time execution based on the critical portions of the model.

A processor and a memory configured to:

identify portions of a model as being either critical to a 2 real-time execution of the model or non-critical to a real-time 3 execution of the model, and

generate code that is capable of real-time execution based on 5 the critical portions of the model. 6

10. A method comprising:

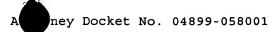
specifying a model, the model including sections, a first subset of the sections designated post-processing unit sections and a second subset of the sections designated as core processing unit sections; and

generating software source code for the model with a code generator using the second subset.

- The method of claim 10 wherein the post-processing unit sections are logical units of the model that have no data outputs that feed core processing unit sections.
- The method of claim 10 further comprising: 12. 1

linking the code to the first subset of sections through an 2 inter-process communication link; and 3 executing the code on a target processor.

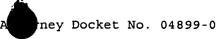
The method of claim 10 wherein specifying the model comprises 1 receiving a user input through a graphical user \interface (GUI). 2



- The method of claim 10 wherein generating comprises applying 1
- 2 a set of software instructions resident in the code generator to
- the second subset.
- The method of claim 12 further comprising: 1
- receiving output from the code via the inter-process
- communication's link; and 3
- processing the output in the first subset.
- A system comprising a graphical user interface (GUI) adapted 1
- to receive user inputs to specify components of a model, the 2
- components containing a first subset of sections designated as 3
- post-processing elements of a model and a second subset of 4
- sections designated as\core elements of the model. 5
- 17. The system of claim \16 further comprising an automatic code
- generator to generate code\capable of real-time execution based on 2
- the second subset of the sections.
- The system of claim 17 wherein the second subset includes 1
- elements representing essential \computational components of the 2
- model. 3
- The system of claim 16 further\comprising a link to provide 1
- inter-process communication between the code and the first subset 2
- of sections of the model. 3
- The system of claim 19 wherein the first subset is non-real 20. 1
- time post-processing sections. 2
- The system of claim 16 wherein the automatic code generator 21. 1
- comprises a set of pre-defined instructions resident in the 2
- automatic code generator to generate code corresponding to the 3
- second subset.

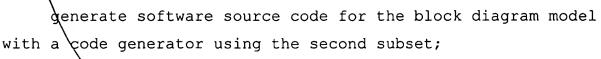
7

8



- The system of claim 21 wherein the code is C programming 1
- 2 languagė.
- The system of claim 16 further comprising a compiler for 1
- compiling the code for a target processor. 2
- A method comprising: 1
- receiving user input through a graphical user interface (GUI) 2
- specifying a block diagram model, the block diagram model 3
- including sections, a first subset of the sections designated 4
- post-processing unit sections and a second subset of the section 5
- designated as core\processing unit sections; 6
 - generating software source code for the block diagram model with a code generator using the second subset;
 - linking the software source code to the first subset via an inter-process communication link; and
 - compiling the software source code into executable code.
 - The method of claim 24 \further comprising executing the executable code on a target processor.
- A computer program product residing on a computer readable
- medium having instructions stored\thereon which, when executed by 2
- 3 the processor, cause the processor to:
- specify a model, the model including sections, a first subset 4
- of the sections designated post-processing unit sections and a 5
- second subset of the section designated as core processing unit 6
- sections; and 7
- generate software source code for the model with a code 8
- generator using the second subset. 9
- The computer program product of claim 26 wherein the computer 1
- readable medium is a random access memory (RAM). 2

- 1 28. The computer program product of claim 26 wherein the computer
- readable medium is read only memory (ROM).
- 29. The computer program product of claim 26 wherein the computer
- 2 readable medium is hard disk drive.
- 30. A processor and a memory configured to:
- specify a\block diagram model, the block diagram model
- including data having internal pre-defined data storage classes
- and external custom data storage classes; and
- generate software source code for the block diagram model
- 6 with a code generator using the internal predefined data storage
- classes and the exterhal custom data storage classes.
- 1 31. The processor and memory of claim 30 wherein the processor
- and the memory are incorporated into a personal computer.
 - 32. The processor and memary of claim 30 wherein the processor
 - and the memory are incorporated into a network server residing in
 - 3 the Internet.
 - 1 33. The processor and memory of claim 30 wherein the processor
- and the memory are incorporated into a single board computer.
- 1 34. A computer program product restaing on a computer readable
- medium having instructions stored thereon which, when executed by
- the processor, cause the processor to:
- receive user input through a graphical user interface (GUI)
- specifying a block diagram model, the block diagram model
- including sections, a first subset of the sections designated
- post-processing unit sections and a second subset of the section
- 8 designated as core processing unit sections; and



link the software source code to the first subset via an inter-process communication link; and compile the software source code into executable code.

35. A processor and a memory configured to:

receive user input through a graphical user interface (GUI) specifying a block diagram model, the block diagram model including sections, a first subset of the sections designated post-processing unit sections and a second subset of the section designated as core processing unit sections; and

generate software source code for the block diagram model with a code generator using the second subset;

link the software source code to the first subset via an inter-process communication link; and compile the software source code into executable code.